



UNITED STATES PATENT AND TRADEMARK OFFICE

45
UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/924,259	08/07/2001	Daniel Lyakovetsky	MM4459	4902

7590

07/20/2005

CHARLES N J RUGGIERO ESQ
OHLANDT GREELEY RUGGIERO & PERLE LLP
ONE LANDMARK SQUARE 10th FLOOR
STAMFORD, CT 06901-2682

EXAMINER

LY, ANH

ART UNIT	PAPER NUMBER
----------	--------------

2162

DATE MAILED: 07/20/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary

Application No.

09/924,259

Applicant(s)

LYAKOVETSKY ET AL.

Examiner

Anh Ly

Art Unit

2162

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 12 May 2005.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-33 is/are pending in the application.
- 4a) Of the above claim(s) 4, 5 and 8 is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-3, 6, 7 and 9-33 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on 07 August 2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- * See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- 1) ☒ Notice of References Cited (PTO-892)
- 2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
- 3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)
Paper No(s)/Mail Date _____
- 4) ☐ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____
- 5) ☐ Notice of Informal Patent Application (PTO-152)
- 6) ☐ Other: _____

DETAILED ACTION

1. This Office Action is response to Applicants' Amendment filed on 05/12/2005.
2. Claims 4-5 & 8 were cancelled (date 04/15/2004).
3. Claims 26-33 have been added.
4. Claims 1-3, 6-7 and 9-33 are pending in this application.

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. This application currently names joint inventors. In considering patentability of the claims under 35 U.S.C. 103(a), the examiner presumes that the subject matter of the various claims was commonly owned at the time any inventions covered therein were made absent any evidence to the contrary. Applicant is advised of the obligation under 37 CFR 1.56 to point out the inventor and invention dates of each claim that was not commonly owned at the time a later invention was made in order for the examiner to

Art Unit: 2162

consider the applicability of 35 U.S.C. 103(c) and potential 35 U.S.C. 102(e), (f) or (g) prior art under 35 U.S.C. 103(a).

7. Claims 1-3, 6-7 and 9-33 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 6,549,216 issued to Schumacher et al. (hereinafter Schumacher) in view of Pub. No.: US 20030154172 A1 of Guyan et al. (hereinafter Guyan).

With respect to claim 1, Schumacher teaches a task is being accomplished to revise an entrain a database of file records (the occurrence of the events that result from user interaction are to be detected and identified are stored in a file: col. 4, lines 20-32 and col. 7, lines 15-20; also see col. 1, lines 7-15 and abstract);

recording in memory, a response to said event, a graphical user interface of a computer to form one or more emulated responses to said event, wherein said one or more emulated responses (the events are recorded via GUI and events are emulated as a sequence of events with the user interaction sequence: col. 2, lines 14-62, col. 4, lines 60-67 and col. 1-57);

selecting a batch of file records that require said task to be performed to execute changes and/or revisions from a database of file records (selecting event from GUI as shown in fig. 2, col. 5, lines 20-57, abstract and col. 2, lines 18-42 and col. 4, lines 60-67 and col. 5, lines 1-15);

loading a specified task into a computer (recoded event would be retrieved via the loader button if fig. 2, item 216: col. col. 5, lines 45-50 and fig. 2); and

executing said task on said selected batch of file records by matching a member of a given event (executing the selected events or files with the options shown on fig.2: col. 6, lines 15-35 and col. 5, lines 20-35 and lines 67-67).

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not explicitly teach identifying an occurrence of an event that occurs while a task is being accomplished to revise an entry in a database of file records and repeating said identifying and said recording to form a collection of emulated event handlers corresponding to events that may occur during said task.

However, Guyan teaches identifying events that occurs in the life of a claim (sections 1304, 1641 and 1643, also see fig. 14); event handlers for the appropriate event (section 0171, 0173 and 0190); collection of event such as claims or tasks (section 0231); and making the tentative changes of the objects (sections 0242, 0322 and 1589).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15

Art Unit: 2162

and fig. 21), would incorporate the use stored events and the event queue waiting for response from the user, in the same conventional manner as discussed by Guyan (sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

With respect to claim 2, Schumatcher teaches a process for automatically revising data as discussed in claim 1.

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not explicitly teach collection of emulated event handlers until an event occurs that indicates that said task is completed.

However, Guyan teaches handling claim processing (event/task) until the task is complete (sections 0122, 0125 and 0127-128).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for

response from the user, in the same conventional manner as discussed by Guyan (sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

With respect to claim 3, Schumatcher teaches a process for automatically revising data as discussed in claim 1.

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not explicitly teach collection of emulated event handlers until an event occurs that indicates that said task is completed.

However, Guyan teaches handling the events task deemed unsuccessful (section 1599).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for response from the user, in the same conventional manner as discussed by Guyan

Art Unit: 2162

(sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

With respect to claim 6-7 and 9-10, Schumatcher teaches a process for automatically revising data as discussed in claim 1.

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62). Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not explicitly teach avoid an unnecessary step, to include a variable as a substitute for a fixed value; a plurality of emulated event handlers for handling a plurality of events that may occurs during said task and obtains values for said variable from said file records when handling said event.

However, Guyan teaches avoiding unnecessary step (sections 0093, 0096 and 0166); substitution value (section 0050); handler event for a plurality of events (section 0231); and replacing the value (section 0050).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for

response from the user, in the same conventional manner as discussed by Guyan (sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

With respect to claim 11, Schumatcher teaches a task is being accomplished to revise a file record (the occurrence of the events that result from user interaction are to be detected and identified are stored in a file: col. 4, lines 20-32 and col. 7, lines 15-20; also see col. 1, lines 7-15 and abstract); and

recording in a memory, a response to said event with a graphical user interface (the events are recorded via GUI and events are emulated as a sequence of events with the user interaction sequence: col. 2, lines 14-62, col. 4, lines 60-67 and col. 1-57).

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not explicitly teach identifying an occurrence of an event that occurs while a task is being accomplished to revise an entry in a database of file records and storing said response in an emulated event handler.

However, Guyan teaches identifying events that occurs in the life of a claim (sections 1304, 1641 and 1643, also see fig. 14); event handlers for the appropriate event (abstract and sections 0016, 0171, 0173 and 0190) and collection of event such as claims or tasks (section 0231) and making the tentative changes of the objects (sections 0242, 0322 and 1589).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for response from the user, in the same conventional manner as discussed by Guyan (sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

With respect to claims 12-15, Schumatcher teaches a method as discussed in claim 11. Also Schumatcher teaches accessing the event record (col. 3, lines 65-67, col. 4, lines 1-10 and col. 7, lines 20-35).

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks

Art Unit: 2162

and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not clearly teach emulated event handler to handle said event for said obtained record, storing for a plurality of events that occur while accomplishing said task, task for said obtained record is completed.

However, Guyan teaches identifying events that occurs in the life of a claim (sections 1304, 1641 and 1643, also see fig. 14); event handlers for the appropriate event (abstract and sections 0016, 0171, 0173 and 0190) and collection of event such as claims or tasks (section 0231) and making the tentative changes of the objects (sections 0242, 0322 and 1589).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for response from the user, in the same conventional manner as discussed by Guyan (sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

Claim 16 is essentially the same as claim 11 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 11 hereinabove.

Claim 17 is essentially the same as claim 12 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 12 hereinabove.

Claim 18 is essentially the same as claim 13 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 13 hereinabove.

Claim 19 is essentially the same as claim 14 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 14 hereinabove.

Claim 20 is essentially the same as claim 15 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 15 hereinabove.

Claim 21 is essentially the same as claim 11 except that it is directed to a program rather than a method, and is rejected for the same reason as applied to the claim 11 hereinabove.

Claim 22 is essentially the same as claim 12 except that it is directed to a program rather than a method, and is rejected for the same reason as applied to the claim 12 hereinabove.

Claim 23 is essentially the same as claim 13 except that it is directed to a program rather than a method, and is rejected for the same reason as applied to the claim 13 hereinabove.

Claim 24 is essentially the same as claim 14 except that it is directed to a program rather than a method, and is rejected for the same reason as applied to the claim 14 hereinabove.

Claim 25 is essentially the same as claim 15 except that it is directed to a program rather than a method, and is rejected for the same reason as applied to the claim 15 hereinabove.

With respect to claim 26, Schumatcher teaches a process for automatically revising data as discussed in claim 1.

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not wherein said event comprises a message that requires said human operator to determine an appropriate action.

However, Guyan teaches an appropriate action/event (sections 0171 and 0284).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for

Art Unit: 2162

response from the user, in the same conventional manner as discussed by Guyan (sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

With respect to claim 27, Schumatcher teaches a process for automatically revising data as discussed in claim 1.

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not clearly teach wherein said event comprises a message that requires said human operator to provide information to revise said entry.

However, Guyan teaches making changes the records (sections 0322, 0959, 0973 and 1426).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for response from the user, in the same conventional manner as discussed by Guyan

(sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

With respect to claim 28, Schumatcher teaches a method as discussed in claim 11.

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not wherein said event comprises a message that requires said human operator to determine an appropriate action.

However, Guyan teaches an appropriate action/event (sections 0171 and 0284).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for response from the user, in the same conventional manner as discussed by Guyan (sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the

operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

With respect to claim 29, Schumatcher teaches a method as discussed in claim 11.

Schumatcher teaches a GUI-oriented system like windows for processing an emulated sequence of events via user interaction for recoding, selecting, loading, executing and identifying an event with its responses from the stored even file (see abstract and col. 2, lines 14-62), detecting the occurrence of the event such as a clicking on the button with the mouse and response to the event. Schumatcher teaches event handling as a way of detecting and processing user input such as mouse clicks and key presses and iconifying windows (col. 1, lines 52-55). Schumatcher does not clearly teach wherein said event comprises a message that requires said human operator to provide information to revise said entry.

However, Guyan teaches making changes the records (sections 0322, 0959, 0973 and 1426).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to combine the teachings of Schumatcher with the teachings of Guyan, wherein the control unit provided therein (Schumatcher's fig. 15 and fig. 21), would incorporate the use stored events and the event queue waiting for response from the user, in the same conventional manner as discussed by Guyan (sections 1304, fig. 14, sections 0171 and 0242). The motivation being to reduce the

Art Unit: 2162

operator responses, eliminate the unnecessary actions and reduce the overhead in servicing response recording in the memory.

Claim 30 is essentially the same as claim 28 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 28 hereinabove.

Claim 31 is essentially the same as claim 29 except that it is directed to a system rather than a method, and is rejected for the same reason as applied to the claim 29 hereinabove.

Claim 32 is essentially the same as claim 28 except that it is directed to a program rather than a method, and is rejected for the same reason as applied to the claim 28 hereinabove.


Claim 33 is essentially the same as claim 29 except that it is directed to a program rather than a method, and is rejected for the same reason as applied to the claim 29 hereinabove.

Contact Information

8. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Anh Ly whose telephone number is (571) 272-4039 or via E-Mail: ANH.LY@USPTO.GOV or fax to **(571) 273-4039**. The examiner can normally be reached on TUESDAY – THURSDAY from 8:30 AM – 3:30 PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, John Breene, can be reached on (571) 272-4107 or **Primary Examiner Jean Corrielus (571) 272-4032**.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). Any response to this action should be mailed to: Commissioner of Patents and Trademarks, Washington, D.C. 20231, or faxed to: Central Fax Center **(571) 273-8300**

ANH LY 
JUL. 15th, 2005


JEAN M. CORRIELUS
PRIMARY EXAMINER